Database Normalisation

- "The process of creating small stable data structures from complex groups of data when designing a relational database"
- "The systematic process of taking a set of unorganized tables with design pitfalls (eg, no integrity, no consistency, redundancy, etc) and re-organising them to remove the problems."
- So what is it?
    - breaking down complex data structures into structured tables for a relational database

Purpose of normalisation

- The process of normalisation is used to arrive at the best possible design for a database and should ensure that:

    - data is not unnecessarily duplicated (redundancy)
    - data is consistent
    - data structures are easier to maintain
    - the structure of each table is flexible enough to allow as many or as few items as required to be stored
        - (eg a student can study as many courses as they want rather than course1, course2, course3)
    - users can make any kind of complex query, including across tables by the use of primary keys and foreign keys

Normal forms

- Measured in what is called **"normal forms"**
    - **0NF** - unnormalised form
    - **1NF** - 1st Normal form = atomic data, no repeating groups of data, primary key (The KEY)
    - **2NF** - 2nd Normal form = 1NF + No partial key dependence (THE WHOLE KEY)   **(applies to tables with compound/composite/complex keys)**
    - **3NF** - 3rd Normal form = 2NF + No non-key dependence (AND NOTHING BUT THE KEY)
- 3NF is good enough for most designs

Process of normalisation

- eliminate **repeating groups of data and non-atomic data** (1NF)
- eliminate **redundant data depending on part of a primary key** (2NF)
- eliminate **redundant data dependent on non-primary key fields** (3NF)

Unnormalised form (0NF)
- Data has **not** been normalised
- Table structure may contain these characteristics:
    - non-atomic data (a field may contain more than one data item per record)
    - repeating groups of data (some fields may be repeated for each record)
    - No Primary key
    - Calculated fields (this may be ok for historical data such as shop sales).

Non-atomic data

> a field may contain more than one data item per record

Non-Atomic Data

ID Number	Name
0192	Tony Blair
2345	Margaret Thatcher
2201	Winston Churchill
8420	John Major

Repeating groups of data

> One or more fields are repeated for each record.

Repeating groups of data

ID	Food	Ingredient	Ingredient	Ingredient
1	Chocolate	Cocoa	Colouring	Milk
2	Coffee	Coffee Beans		
3	Sponge	Flour	Egg	Milk

More repeating groups of data

> One or more fields are repeated for each record.

Repeating groups of data

Order No	Date	Product	Quantity	Price
4828	3/9/13	Pens	4	£0.49
		Air Horn	1	£3.50
4829	4/9/13	Fuji HS30EXR	1	£389.00
		Coffee	3	£1.84
		PC Pro Dec 13	1	£3.99
4830	4/9/13	Galaxy S3	1	£549.99

Why is this table in 0NF?

Non-Atomic Data

Surname	Forename	Address
Blair	Tony	5 Fairfax Road Sutton Coldfield B75 7IP

Why is this table in 0NF?

Non-Atomic Data

Product Code	Description	Price
ABC123	Blue Sandal Size 9	£13.99

Why is this table in 0NF?

Non-Atomic Data

PC Code	Make	Repair Date
005010	Stone	5/3/12, 12/6/12, 16/3/13
005004	Dell	12/6/12
005021	Stone	2/4/12, 29/9/12

This could potentially be considered to be repeating groups of data too.

Putting the table into 1NF

data is now atomic

ID Number	Surname	Forename
0192	Blair	Tony
2345	Thatcher	Margaret
2201	Churchill	Winston
8420	Major	John

Putting the table into 1NF

> Create a separate record for each ingredient

ID	Food	Ingredient
1	Chocolate	Cocoa
1	Chocolate	Colouring
1	Chocolate	Milk
2	Coffee	Coffee Beans
3	Sponge	Flour
3	Sponge	Egg
3	Sponge	Milk

PK?? So strictly not 1NF. Many to Many link

Characteristics of 1NF:
- All data must be ATOMIC
  - one data item per field
- No repeating groups of data
  - no sets of repeating fields
  eg no course1, course2, course3

Unique PK and (sometimes) no calculated fields

Second normal form (2NF)

KEY LEARNING POINT

Characteristics of 2NF:

> Data must be in 1NF *(easy to remember!)*

> All non-primary key attributes MUST be dependent on the WHOLE primary key
   - where the primary key is a composite key (more than one field)

NO PARTIAL KEY DEPENDENCIES

Dependency:

> B is dependent on A if once you know A you can determine B
   - For a given value of A there is only one possible value of B associated with it

   - TABLE (attribute A, attribute C, attribute1, attribute2, attribute B)

   partial key

Identifying table not in 2NF

Primary Key. Student name is only dependent on Student ID – which is only PART of the primary key.

Course Code	Student ID	Student Name	Grade
CS1	9281	Blair	B
CS1	3819	Thatcher	D
Dr91	1039	Churchill	A
Dr91	2910	Major	E
Dr91	3819	Thatcher	C

Putting a table into 2NF

Remove the data dependent on only student ID (partial key)

Create a new table called **STUDENT**.

There are now no partial key dependencies.

Course Code	Student ID	Grade
CS1	9281	B
CS1	3819	D
Dr91	1039	A
Dr91	2910	E
Dr91	3819	C

Student ID	Student Name
9281	Blair
1039	Churchill
2910	Major
3819	Thatcher

Student ID becomes a foreign key

1) Explain why the data structure in the following table is not in First Normal Form.

Student Name	Year	Classes	Games	Games
Fred Smith	11	11ICT 10Ma2	Football	Rugby
Sonia Banks	11	11MA4, 11Mu	Netball	Lacrosse
Maggie Stuart	10	11En1, 11Ma1	Hockey	Rugby

2) Explain why the data structure in the following table is not in Second Normal Form

Seat Number*	Performance	Date*	Time*	Customer Forename	Customer Surname
A1	We Will Rock You	27/12/2004	2.30	John	May
B1	We Will Rock You	27/12/2004	2.30	Fred	Deacon
C1	We Will Rock You	27/12/2004	2.30	Brian	Taylor
A1	We Will Rock You	27/12/2004	7.30	Roger	Mercury

(The primary key is a compound key- Seat Number, Date and Time)

Partial

Third normal form (3NF)

KEY LEARNING POINT

Characteristics of 3NF:

➢ Data must be in 2NF *(easy to remember!)*

➢ All non-primary key attributes MUST be dependent on the primary key
 • no attributes can be dependent on any **non-key** attribute

NO NON-KEY DEPENDENCIES

Dependency:

➢ B is dependent on A if once you know A you can determine B
 • For a given value of A there is only one possible value of B associated with it

 • TABLE (attribute 1, attribute 2, attributeA, attributeB, attribute C)

non-primary key

Identify table not in 3NF

Primary Key — Film Name and Duration are dependent on Film ID which is NOT a Primary Key

Screen No	Time	Film ID	Film Name	Duration
1	16:00	HP	Harry Potter	120
1	20:00	BR	Blade Runner	115
2	19:30	Tin	Titanic	195
3	19:30	PoA	Planet of the Apes	180
4	20:00	PoA	Planet of the Apes	180

Non-Key attribute

Putting a table into 3NF

Remove the data dependent on only Film ID (non-key)

Create a new table called **FILM**.

There are now no non-key dependencies.

Screen No	Time	Film ID
1	16:00	HP
1	20:00	BR
2	19:30	Tin
3	19:30	PoA
4	20:00	PoA

Film ID	Film Name	Duration
HP	Harry Potter	120
BR	Blade Runner	115
Tin	Titanic	195
PoA	Planet of the Apes	180

Film ID becomes a foreign key

3) Explain why the data structure in the following table is not in Third Normal Form

ID*	Forename	Surname	HouseName/number	Town	County	Postcode
1	John	Davies	18 Bright Road	Nottingham	Notts	NG8 5EP
2	Alice	Hall	24 halls Avenue	Nottingham	Notts	NG8 5ET
3	Joan	Stevenson	19 Walbrook Close	Nottingham	Notts	NG8 5EZ

Question 4

A human resources department stores details of staff in two flat files. One file is used to keep the staff details, and the other file is used to keep records of training undertaken by members of staff. The file structures are as shown below.

Staff_File(Staff-number, Forename, Surname, Address, Home-phone-no, Start-date, Department, Extension-no)

Training_File(Staff-number, Forename, Surname, Department, Training-course, Training-date)
Underlined fields are **key fields** in each table.

(a) Describe, using examples from the above two tables, two problems of these file structures.

(6 marks)

(b) Design an efficient database structure for the above data. (4 marks)

The Answer

4(a). Look at the key field in the training file and think about what **key** means. If staff number was a primary key in this file staff would only be able to attend one course! The other problem is that you are duplicating the staff forename and surname **already** stored in the staff file – this increases **redundancy** and increases the chances of **inconsistency**.

Key words are underlined.

4(b). Try and identify the type of relationship first! In this case it's many-to-many i.e. a training-course can contain many members of staff but one member of staff could attend many courses. A link table resolves the relationship.

tblStaff (<u>Staff-number</u>, Forename, Surname, Address, Home-phone-no, Start-date, Department, Extension-no)

tblTraining (<u>Training-courseID</u>, Training-course, Training-date)

tblStaff-Training (<u>Staff-number</u>, <u>Training-courseID</u>)

It has been suggested that Department warrants a separate table and in a fully normalised database this is true in which case the database would be:

tblStaff (<u>Staff-number</u>, Forename, Surname, Address, Home-phone-no, Start-date, DepartmentID, Extension-no)

tblTraining (<u>Training-courseID</u>, Training-course, Training-date)

tblStaff-Training (<u>Staff-number</u>, <u>Training-courseID</u>)

tblDepartment (<u>DepartmentID</u>, Department)

NB: I personally wouldn't use most of AQAs names in a real database (e.g. Home-phone-no should be Home_phone_no or HomePhoneNo). The format is only preserved to make the answer relate to the question.   (FatMax).

Questions on Relational databases and normalisation (HW 3)

1. A boat hire company stores details of its customers, boats and rentals in a database. The relations in the database hold the following data:

Boat (<u>BoatName</u>, type, length, berths)

Customer (<u>CustomerID</u>, Surname, Firstname, AddressLine1, AddressLine2, Town, Postcode, DateOfBirth, email)

BoatHire (<u>CustomerID, BoatName, HireDate</u>, HireEndDate)

(a) The key in the BoatHire relation consists of three attributes.

What is the name given to a key made up of more than one attribute? [1]

_____

(b) The relations in this database are in Third Normal Form.
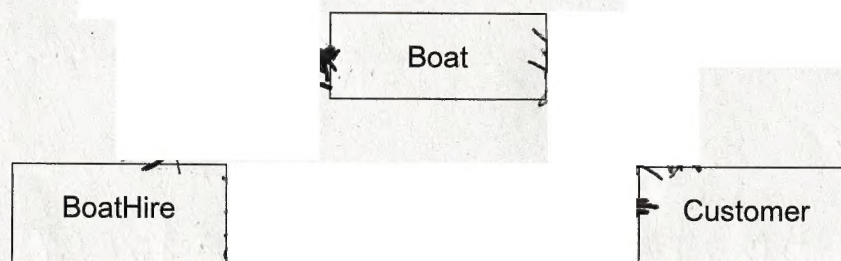
State the properties of a relation in Third Normal Form. [3]

(c) State, with reasons, why it is important that relations in a database are in Third Normal Form. [6]

(d) Complete the Entity-Relationship diagram below to show the degree of the relationships between the entities. [3]

Boat

BoatHire

Customer

2. A database is to be created to hold data about students at a Sixth Form College and the subjects they study.

Students study a number of subjects, and each subject has one subject leader.

The table below is a first attempt at the design of the database.

StudentID	Name	TutorGroup	Tutor	Subject	Level	SubjectLeader
S1000	Bella	2	CKE	History	A	AJH
				Geography	A	BJG
				Economics	AS	CKE
S2000	Jane	3	KPR	English	AS	DRE
				French	A	FJF
				Russian	A	KPR
S3000	Greg	1	DRE	English	A	DRE
				Geography	A	BJG

(a) Explain, with reference to the data, why the table is not in First Normal Form (1NF). [2]

(b) The design is changed to:

Student (StudentID, Name, TutorGroup, Tutor)

StudentSubject (StudentID, Subject, Level, SubjectLeader)

Show how the data given in the table above would be held in these two tables. [3]

Table: **Student**

StudentID	Name	TutorGroup	Tutor

Table: **StudentSubject**

StudentID	Subject	Level	SubjectLeader

(c) A student is not allowed to study the same subject at A and AS Level. [1]

What is the primary key of the table StudentSubject? _____

(d) The two tables Student and StudentSubject are related. Explain how this is achieved using a primary and a foreign key. [2]

(e) Explain why the table StudentSubject is not in Second Normal Form (2NF). [2]

(f) Explain why the table Student is not in Third Normal Form (3NF) [2]

[Total 25 Marks]

Questions on Introduction to SQL (Hw 4)

1. A second-hand car dealer keeps a record of customers who have purchased cars. The data held on each table includes the following fields:

Customer

CustID	Surname	Initials	email
1234	Banya	F	fbanya@hotmail.com
1240	Shreeve	PL	plshreeve@icloud.com
1245	Barker	AD	annBarker@fbarker.com
1266	Moore	C	colinm129@gmail.com
1304	Horton	JJ	jjh@yahoo.co.uk
1366	Gold	DS	dsg@yahoo.co.uk

Car

ID	Make	Model	YearRegistered	Mileage	Doors	Price	CustID	PurchaseDate
F03	Ford	Fiesta	2000	116,000	3	£999	1304	12/04/2004
F05	Ford	Mondeo	2005	92,000	5	£1,200	1240	05/07/2009
N07	Nissan	Juke	2012	33,400	3	£5,500	1366	16/11/2015
N09	Nissan	Micra	2001	92,500	3	£895	1366	30/05/2008
P08	Peugeot	407 Estate	2008	56,700	5	£2,850	1245	01/06/2011
V01	Vauxhall	Meriva	2005	105,000	5	£1,020	1366	22/04/2008
V02	Vauxhall	Insignia	2012	56,000	5	£9,000	1234	10/10/2013
V04	VW	Beetle	2003	155,000	3	£995	1234	03/08/2008
V06	Vauxhall	Astra	2009	30,000	5	£5,395	1240	17/09/2011

(a) Write the entity descriptions for the entities **Customer** and **Car** in the format:

Entity(attribute1, attribute2, ….) [2]

(b) Identify primary and foreign keys, where they exist, in both tables. [2]

(c) Draw an entity relationship diagram showing the relationship between the two tables. [2]

(d) Write an SQL statement using keywords SELECT... FROM... WHERE...ORDER BY to extract a list of the make, model, year registered and price of all 5-door cars costing less than £2000, sequenced in descending order of price. [4]

(e) Write an SQL statement using keywords SELECT... FROM... WHERE...ORDER BY to extract a list, sequenced by surname and initials, of customer IDs, surnames, initials and email addresses of all customers who purch   veen 2009 and 2011 inclusive, together with the make of car and price paid. [6]

(f) Write an SQL statement to extract all the details of cars with *make* beginning with "V". [2]

(g) The Car database is in Third Normal Form. Explain what this means. [2]

Total 20 marks